

## 周报/梅鸿辉

### 8.1-8.7

## VisComposer 相关

重新阅读了一遍类似工作的论文（D3[1], VEGA[2], Iyra[4], protoVis[5], iVisDesigner[6], iVoLVER[7]等），考虑了下还可以有所工作的方向。

VEGA 本身基于 D3, 在树状的基本图元定义之上添加了基于 *signal/predicates/scale inversion* 的事件响应机制用于定义可视化交互（在[3]中有详细的阐述）。为了达到更好的运行效率以支持动态交互，引入了 *changesets* 的概念。总体来说，初看时觉得是很简单的思路，但自己视图模仿实现时发现了很多不容易解决的问题，回头再看发现论文中其实都有阐述。

iVisDesigner 其实主体还是定制的模板+交互。我认为其核心思路还是在于数据与模板的关系。数据并非被复制进模板而是一种双向的、更紧密的关系；这样做一方面提高了性能（引用而非复制），另一方面增加了交互的粘性，还使得视图间的联动成为了天然的特性（基于附着在原始数据之上的 meta data）。

iVoLVER[7]提供了通过交互手段（涂抹、描线）等方法从图像中提取出可视化信息并绘制替代图表的方法。其图表->数据->图表的流程不同于普遍的数据->图表的思路。我希望我们能做成是在自身数据和图表之间不断循环优化的一个过程。

[8]提供了可视化系统的 *storytelling*。给我的感觉是，将可视化交互的范围扩大了（交互反馈的那个箭头直接划到了最前端，而一般的系统只能在一定范围内）。我认为这是可视化制作工具的非常重要的一个作用。用户将不满足于在系统作者控制的范围内进行自行探索，而可以从更底层发掘信息。

inMens[9]提供了一些基于 datacube 的简单图表。本论文主体是 datacube 的运算，图表只是附属。对于 VisComposer 来说：可以考虑增加后端内容；同时应能引导用户创建更加符合数据特性的图表。并非是类似 tableau 里基于数值类型的一次性推荐，而是在整个交互创建的过程中不断发掘、调整的完整路径。

### 零碎的想法

- ◆ 所有工具都是数据->图表，能否反向创建？
- ◆ 视觉绑定对于复杂图形（path）非常难以自动识别可以绑定的属性。无法做到任意导入的情况下，提供一些简单的绘制工具，将结果图形限制在系统提供的框架中。可绑定属性可以根据用户在创作过程中的操作步骤发掘，因为图形的核心属性必然与其绘制步骤息息相关（例如，画圆是从圆心拖拽一个半径得到，因为圆最重要的属性就是圆心和半径）。
- ◆ 现有系统无法原生处理递归式的结构，只能依赖现有模板（包括 D3，一般先使用 layout 进行计算；即使自行递归调用，事实上已经脱离了 D3 的体系属于编程的灵活性部分）。

可以考虑在这一点上寻找解决方案。

- ◆ 关于交互。交互的触发在图形空间中，但 VEGA 和 iVisDesigner 都是在数据空间进行交互的后续操作（过滤、选取、位置变换等）。对于一些 distortion 类的变换这样做是难以完成了。目前正在寻找 D3 中相关话题的代码看看他们是如何实现的。
- ◆ 关于[7][8][9]的思考（见上文）。创建过程体现需完整的思路：自顶向下、逐步细化的创建图表。类似 VisComposer 视频中的散点图->直方图->折线图的变化。

## 论文阅读

### VisComposer 相关

1. M. Bostock, V. Ogievetsky, and J. Heer, “D3: Data-Driven Documents,” *IEEE TVCG*, vol. 17, no. 12, pp. 2301–2309, Dec. 2011.
2. A. Satyanarayan, R. Russell, J. Hoffswell, and J. Heer, “Reactive Vega: A Streaming Dataflow Architecture for Declarative Interactive Visualization,” *IEEE TVCG*, vol. 22, no. 1, pp. 659–668, 2016.
3. A. Satyanarayan, K. Wongsuphasawat, and J. Heer, “Declarative interaction design for data visualization,” *Proc. 27th Annu. ACM Symp. User interface Softw. Technol. - UIST ’ 14*, pp. 669–678, 2014.
4. A. Satyanarayan and J. Heer, “Lyra: An Interactive Visualization Design Environment,” *Comput. Graph. Forum*, vol. 33, no. 3, pp. 351–360, Jun. 2014.
5. M. Bostock and J. Heer, “Protovis: A Graphical Toolkit for Visualization,” *IEEE TVCG*, vol. 15, no. 6, pp. 1121–1128, Nov. 2009.
6. D. Ren, T. Hollerer, and X. Yuan, “iVisDesigner: Expressive Interactive Design of Information Visualizations,” *IEEE TVCG*, vol. 20, no. 12, pp. 2092–2101, Dec. 2014.
7. G. G. Méndez, M. A. Nacenta, and S. Vandenheste, “iVoLVER: Interactive Visual Language for Visualization Extraction and Reconstruction,” in *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems - CHI ’ 16*, 2016, pp. 4073–4085.
8. S. Gratzl, A. Lex, N. Gehlenborg, N. Cosgrove, and M. Streit, “From Visual Exploration to Storytelling and Back Again,” vol. 35, no. 3, pp. 1–18, 2016.
9. Z. Liu, B. Jiang, and J. Heer, “imMens : Real-time Visual Querying of Big Data,” *Comput. Graph. Forum*, vol. 32, no. 3pt4, pp. 421–430, Jun. 2013.
10. S. Gratzl, A. Lex, N. Gehlenborg, N. Cosgrove, and M. Streit, “From Visual Exploration to Storytelling and Back Again,” vol. 35, no. 3, pp. 1–18, 2016.

### 其他

11. T. Wu, Y. Wu, C. Shi, H. Qu, and W. Cui, “PieceStack: Toward Better Understanding of,” *IEEE TVCG*, vol. 22, no. 6, pp. 1–1, 2016.

巫老师小组的文章。我主要是想看看一些 stacked steaming graph 可能用到的图形、变化和

交互。

12. J. Johansson and C. Forsell, "Evaluation of Parallel Coordinates: Overview, Categorization and Guidelines for Future Research," *IEEE TVCG*, vol. 22, no. 1, pp. 579–588, 2016.

各种 PC 方法的介绍和评估，还与其他许多方法如散点图等进行了比较。